

# Pràctiques integrades

## *Pràctica 12*

Matemàtiques, curs 2001-2002.

### 12 Geometria del pla

El paquet **geometry** permet definir i treballar amb objectes geomètrics del pla euclidià.

```
> with(geometry):
```

#### 12.1 Punts i rectes.

L'objecte més senzill que podem definir i utilitzar amb el paquet **geometry** és un punt. Per a definir un punt utilitzem la funció `point(p,x,y)`, on `p` és el nom del punt, `x` és la primera coordenada i `y` la segona.

##### Exemple 12.1

Si volem definir els punts  $A=(1,1)$ ,  $B=(2,2)$  i  $C=(-1,1)$  posarem:

```
> point(A,1,1);point(B,2,2);point(C,-1,1);
```

El segon objecte que podem definir és una recta, mitjançant la funció **line**. Hi ha dues maneres d'introduir una recta, mitjançant dos punts per on passi, o mitjançant una equació:

##### Exemple 12.2

Podem definir la recta `l1` com la que passa pels punts `A` i `B` que hem definit abans:

```
> line(l1,[A,B]);
```

També podem definir la recta `l2` a partir d'una equació, per exemple  $x + y = 12$

```
> line(l2,x+y=12,[x,y]);
```

El tercer dels arguments de la comanda **line** anterior fa referència al nom que tenen les coordenades dels eixos amb els que treballem. Això no està prefixat i els podem canviar quan volguem. A la comanda anterior automàticament s'ha assignat el nom `x` a l'eix horitzontal i `y` al vertical.

Si volem, podem forçar globalment el nom que volem per a les coordenades redefinint les variables `_EnvHorizontalName` i `_EnvVerticalName`.

Un cop hem definit i posat nom a aquests objectes, podem demanar les propietats que té un d'ells a partir de les funcions:

- `coordinates(p)`; que retorna les coordenades d'un punt `p` com una llista.
- `form(l)`; que retorna la forma de l'objecte geomètric, (i.e., `line2d` si `l` és una recta).
- `Equation(l)`; que retorna l'equació que defineix la recta `l`.
- `detail(l)`; que torna una descripció detallada de la recta `l`.

**Exemple 12.3**

Redefinim els eixos com  $u$  i  $v$  i demanem les propietats de la recta  $l1$ :

```
> _EnvHorizontalName := u: _EnvVerticalName := v:
> form(l1);
> Equation(l1);
> detail(l1);
```

Quan tenim definides vàries rectes i punts podem demanar informació de la posició relativa que tenen i fer càlculs mitjançant:

- `AreParallel(l1,l2)`; que retorna `True` si les rectes són paral·leles i `False` si no ho són.
- `ArePerpendicular(l1, l2)`; que retorna `True` si les rectes són perpendiculars i `False` si no ho són.
- `AreConcurrent(l1,l2,l3)`; que retorna `True` si les tres rectes tenen un punt en comú.
- `FindAngle(l1,l2)`; que retorna l'angle que formen dues rectes.
- `intersection(p,l1,l2)`; que guarda a la variable  $p$  la intersecció entre les dues rectes.
- `ParallelLine(l,p,l1)`; que guarda a la variable  $l$  la recta que passa per  $p$  i és paral·lela a  $l1$ .
- `PerpendicularLine(l,p,l1)`; que guarda a la variable  $l$  la recta que passa per  $p$  i és perpendicular a  $l1$ .
- `projection(q, p, l)`; que guarda a la variable  $q$  el punt que resulta de fer una projecció perpendicular de  $p$  a la recta  $l$ .
- `distance(p,l)`; que calcula la distància entre un punt  $p$  i una recta  $l$  o bé entre 2 punts.

**Exemple 12.4**

```
> AreParallel(l1,l2);
> ArePerpendicular(l1,l2);
> FindAngle(l1,l2);
> intersection(E,l1,l2);
> detail(E);
> intersection(l,l1,l1);
> detail(l);
```

**Exercici 12.1**

Considereu els punts  $A = (2, -3)$ ,  $B = (1, 8)$  i  $C = (5, -1)$ .

- Determineu l'equació de la recta que passa per  $C$  i és perpendicular a la recta que passa per  $A$  i  $B$ .
- Quin és el punt d'intersecció d'aquestes dues rectes?

- (c) Comproveu si el punt  $D = (-4, 3)$  és d'alguna de les dues rectes anteriors.
- (d) Calculeu la distància que hi ha entre els punts  $A$ ,  $B$ ,  $C$  i  $D$ .
- (e) Quina distància hi ha des del punt  $C$  a la recta que passa per  $A$  i  $B$ ?

Finalment, tots els objectes es poden representar gràficament utilitzant la comanda `draw`.

### Exercici 12.2

Dibuixeu els punts i les rectes de l'exercici anterior.

## 12.2 Còniques.

A part de punts i rectes, també podem treballar amb objectes més complicats com són les còniques. La comanda del paquet **geometry** que permet definir una cònica és `conic()`, que es pot utilitzar de tres formes diferents:

- `conic(p, [A,B,C,E,F], n)`; on la llista `[A,B,C,E,F]` són cinc punts pels quals volem que passi la cònica.
- `conic(p, [dir,foc, exc], n)`; on `dir` ha de ser una objecte `line` que serà la recta directriu de la cònica, `foc` és un objecte `point` que representa un focus de la cònica i `exc` és un valor numèric que representa l'excentricitat.
- `conic(p, eqn, n)`; on `eqn` és una equació quadràtica en dues variables.

En els tres cassos l'argument `p` és el nom amb el que ens referirem a l'objecte que estem definint i l'argument `n` és opcional i és el que determina el nom de les variables respecte les que volem treballar (com en els punts i rectes).

**Nota:** Una cònica es pot definir com el lloc geomètric dels punts  $P$  tals que el quocient de la distància de  $P$  a una recta fixada (la directriu de la cònica) amb la distància de  $P$  a un punt donat (el focus) és una constant (l'excentricitat). Amb aquest punt de vista les el·lipses són les còniques amb excentricitat menor que 1, les paràboles les d'excentricitat 1 i les hipèrboles les d'excentricitat més gran que 1. Les circumferències corresponen al cas degenerat d'excentricitat 0 amb directriu a *l'infinit*.

### Exemple 12.5

```
> eq:=13*x^2 + 150*x- 171 -10*x*y -102*y + 13*y^2 = 0;
> conic(c,eq,[x,y]);
> detail(c);

> draw([c,op(foci(c))(color=blue),center(c)(color=green),
> line(eix,foci(c))(color=yellow),
> PerpendicularLine(eix2,center(c),eix)(color=orange)]);
```

Quan tenim una **el·lipse** podem demanar les següents dades:

- Els dos focus: `foci( )`.

- El centre: `center( )`.
- Les longituds dels eixos major i menor: `MajorAxis( )`, `MinorAxis( )`.
- L'equació: `Equation( )`.

**Exemple 12.6**

```
> foci(c);  
> detail(foci(c));  
> simplify(coordinates(foci(c)[1]));  
> coordinates(center(c));  
> MajorAxis(c);
```

Quan tenim una **hipèrbola** podem demanar les següents dades:

- Els dos focus: `foci( )`.
- El centre: `center( )`.
- Les asímptotes: `asymptotes( )`.
- Els vèrtexs: `vertices( )`.
- L'equació: `Equation( )`.

**Exercici 12.3**

Comproveu que l'equació  $17x^2 + 18xy - 7y^2 - 12x + 36y - 12 = 0$  defineix una hipèrbola i determineu les equacions de les seves asímptotes.

Quan tenim una **paràbola** podem demanar les següents dades:

- El focus: `focus( )`.
- La directriu: `directrix( )`.
- El vèrtex: `vertex( )`.
- L'equació: `Equation( )`.

**Exercici 12.4**

Comproveu que l'equació  $9x^2 + 6xy + y^2 - 4x + 12y - 4 = 0$  determina una paràbola i calculeu l'equació de la seva directriu.

### 12.3 Llista d'exercicis.

#### Exercici 12.5

Determineu el radi d'una circumferència que és tangent a la recta  $x + y = 2$  en el punt  $(1, 1)$  i tal que el seu centre està sobre la recta  $2x - 3y = 1$ . Feu un dibuix on estiguin representats els elements d'aquest problema.

#### Exercici 12.6

Definiu la cònica donada per l'equació  $x^2 - y^2 = 0$  i dibuixeu-la.

#### Exercici 12.7

Feu una funció `dibuix` que, a partir d'un objecte cònica `c`, doni com a resultat un dibuix de la cònica amb els seus elements significatius, diferenciant els cassos possibles (focus, centre i eixos per a una el·lipse,...)

#### Exercici 12.8

Apliqueu la funció `dibuix` que heu disenyat en l'exercici anterior a les còniques següents:

$$9x^2 + 12xy + 4y^2 - 34x + 38y + 22 = 0$$

$$-3x^2 - 8xy + 3y^2 - 28x + 6y + 23 = 0$$

$$\frac{29x^2}{3} - 2xy + 7y^2 - \frac{172x}{3} - 72y + \frac{734}{3} = 0$$